College & Career Awareness

Program Area(s): Information Technology

Lesson Plan Title: Creative Project
Students will create a project.

Estimated Time: 2 days or two 50 minute sessions

Primary CTE Pathway(s) Explored: Software Development/Programming and Digital Media

Intended Learning Outcome(s):

- Become acquainted with a wide range of occupations, CTE Pathways, career trends and emerging careers.
- Understand the importance of making career decisions related to career aspirations.
- Identify career and postsecondary education options through investigation of high school to college and career pathways.
- Consider and explore nontraditional career opportunities.
- Explain how academic content knowledge and technical skills are used in various careers.

College and Career Awareness Standard, Objective(s):
Standard 7, Objective 2

Cross Curriculum Integration:

- **21st Century or Interpersonal Soft Skills—critical thinking, collaboration, communication, creativity:**
- **Language Arts:** story telling in the "Play Box" project
- **Math:** connections: at least 2 lesson plans with tie to Math variables, geometry, angles in the Artist project
- **Technology:**

Career Opportunities in the CTE Pathway(s): 3D animation or graphic design, customer service, database, electronics technician or engineer, engineer, freelancer, hardware, networking or system administrator, programmer or software developer, quality assurance (QA), system analyst or tester, repair and fix, technical support (technician or help desk), technical writing, security expert, Webmaster or Web designer

Nontraditional Career Opportunities: Government and public administration, computer programmers, information technology, manufacturing (CNC), game design and developer, software developer, computer programmer, Web developer, software quality assurance, user interface designer, software entrepreneur

STEM Specific Career Opportunities: Game design and developer, software developer, computer programmer, Web developer, software quality assurance, user interface designer, software entrepreneur

Methods (Approach to Teaching and Learning):

- Direct Instruction and Demonstration
- Activity/Inquiry/Practice Centered Instruction
- Problem-based or Project-Based Learning

Materials Needed:

- Computer with Internet connection
- Sharing cards (handout)

Vocabulary:

- Click – pushing the mouse button down and letting it up. (A mouse click on an event handler starts a set of commands.)
- Creativity – using your imagination to make a game or project in your own way.
- Event – a mouse click, key press, or mouse movement that starts a set of commands.
- Event handler – a button, object, or key that have events associated with them.
- Function – a set of command that are given a name that can be used over and over by just calling the name of the function instead of having to write all the commands over and over.
- Parameter – a place in a command that options for the command can be changed (speed in the ?????? command, changing the sound in the "play _____ sound" command)
- Random value – a computer generated number or value usually in a set range (random values are used in games so the play does not know what to expect)
- Variable – a place or name in a command that values can be changed (set play speed to _____ the blank line acts like a variable)

Prior Knowledge Required by Students:

Basic computer skills, knowledge of skills and concepts learned in previous code.org stages and projects.

Start with an Hour of Code

More than 44 million people of all ages have learned an Hour of Code, a one-hour introductory course designed to demystify computer science and show that anybody can learn the basics. Participate during Computer Science Education Week.

Teach One of the Online Courses

Code studio, an online platform of code.org, enables students to learn the basic concepts of computer science as early as kindergarten. Within code studio, a new "Play Lab" has been developed where elementary students can create and send apps or animations directly to a cell phone, just by typing in the phone number. Or, try our general course designed for any educator to host in a classroom, after-school program, or even in a club outside of school.

Inspire Students (Especially Women) to Try Computer Science

Here's a collection of videos, posters, and how-to resources to help inspire students to try computer science. Also, use our educational video library and nominate a student or teacher you know for a code.org award.

Instructional Procedures:

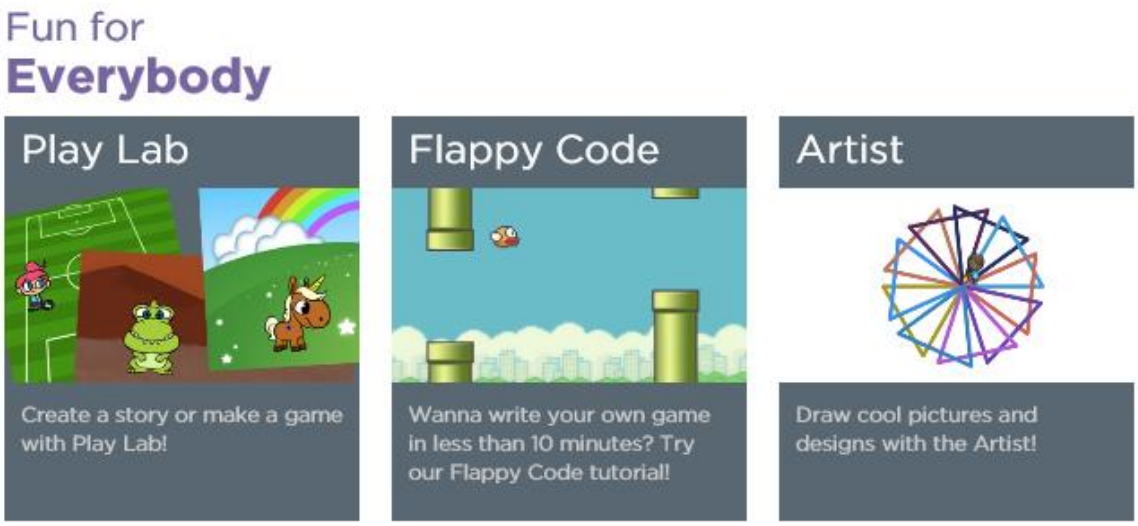
Background

Code.org has three different projects that students can do and experiment with: 1) Flappy Game, 2) Play Lab, and 3) Artist.

1. Flappy Game – this is a project where students learn about events and parameters to create a Flappy Bird type game that adjust in several different ways and then shared with others to play.
2. Play Lab – this project lets the students write a story or create a game that can then be shared with others.
3. Artist – this project allows the students to draw or create designs which can then be shared with others.

All of these projects have a video and about 10 short lessons that lead up to the creative project. In the project stage a student can use their creativity to create a variety of stories, games, design, or drawings. When finished the student is given a Web URL that can be shared with others. The "sharing card" handout can be used by the student to make cards

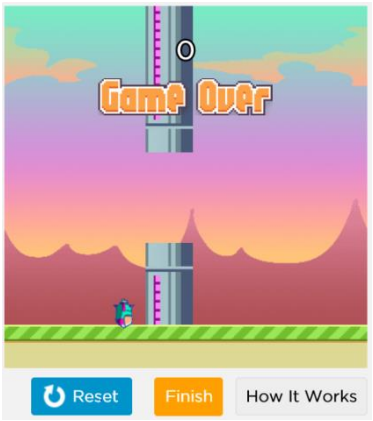
to give to friends, family, parents, and others so they can view/play the student created project. In all of these the student can go back and create additional projects.



Screenshot from code.org

Flappy Game
The Flappy Game lets the students create a Flappy Bird type game that can be shared with others.
To get started go to the URL: http://studio.code.org/flappy/1 or go to http://studio.code.org and select the Flappy Code picture. In this project there are a series of 10 puzzles. They lead up to the 10th puzzle which is the creative puzzle.

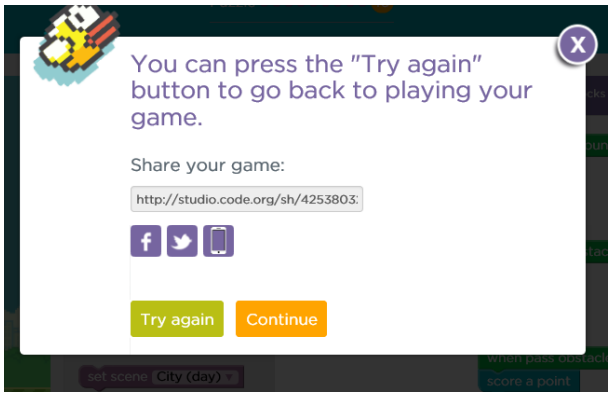
Sample Flappy Game
Sci-Fi sample game http://studio.code.org/sh/42171638



Puzzles Overviews

- **Puzzle 1** – Video with Katie Apone. Katie introduces events and event handlers. When the program is run, the "when clicked" event starts sound and flap.
- **Puzzle 2** – "when hit the ground" event is introduced with the "end game" command.
- **Puzzle 3** – Using "when run" event the "set speed parameter" is introduced. The parameter is a variable that can be adjusted to change the speed.
- **Puzzle 4** – Obstacles are introduced with the "when hit and obstacle" event. Different command can be placed in the event.
- **Puzzle 5** – Score keeping is introduced using the "when pass obstacle" event with the "score point" command under it.
- **Puzzle 6** – Change the flap speed which is introduced with the command, "flap a speed parameter amount". Using the variable in the command the flap speed can be adjusted.
- **Puzzle 7** – "set background parameter" is introduced.
- **Puzzle 8** – Random parameter is introduced. "Random" lets the computer generate or select a value or parameter in a given range. Random commands are used in game design so the user does not always know what to expect.
- **Puzzle 9** – A set command is introduced with the command "set score value". Most computer languages have a way to assign a value to a variable. In this command the word score is a variable. Its value can be changed. Its default is "0" but a person could set the value of score to another value. In math variables usually have a name like "x" or "y" but in computer programming variables usually have a name of what kind of values they store like "score". In many computer languages the set command may look like "score = value" or "score = 5".

- **Puzzle 10** – Create your own Flappy Game. You can change all the visuals and all the rules, even the gravity. When you're done, click "finish" to let friends try your game on their phones, tablets, or computers. Print out the "Code.org Project Sharing Card.docx" handout on business card sheets, card stock, or colored paper and cut into cards for the students.



Code.org Screenshot with link for sharing project

Screenshot from handout "Code.org Sharing Card"

Here the students can set up the Flappy Game in several different ways. They can choose a theme like "Sci-Fi" or "Underwater", etc. In the setup they can change most of the values in the commands to assign new values to variables. One of the real values in education is when students can "publish" work for friends, family, parents, and others instead of just doing work for the teacher. This is excellent way for students to "publish" their creative works.

Play Lab (Optional for the second day)

Play Lab lets you make an interactive game or animation that you can play. In the introduction video a young man introduces several new blocks.

- "when parameter keypress arrow" this event block lets the programmer assign different commands based on the key that is assigned in the block. This event can be followed by the "move actor, actor number parameter direction parameter". This allows the movement of character through the use of keystrokes.
- "when actor, actor number touches actor number" event can be used to assign different commands.
- "when actor, actor number clicked" event can be used to make an actor do additional command.

- **Puzzle 1** – say "words parameter" ' command is introduced.
- **Puzzle 2** – The "actor number" say "words parameter" is introduced.
- **Puzzle 3** – The "move actor number direction distance pixels" command is introduced.
- **Puzzle 4** – "when actor, actor number touches actor number" event is used to have the cat talk to the dog when they touch.
- **Puzzle 5** – The "when actor clicked" event is introduced to have the octopus talk when clicked.
- **Puzzle 6** – The "when left arrow", "when right arrow", "when up arrow", and "when down arrow" events are introduced to allow the movement of characters to accomplish a task.
- **Puzzle 7** – The "repeat forever" loop event block allows you to run code continuously.
- **Puzzle 8** – The "score/remove parameter point" block is introduced which allows the programmer to add or subtract points to the score variable.
- **Puzzle 9** – The challenge is given. Can you add blocks to change the background and the speed of the penguin, and then move him with the arrows until you score?
- **Puzzle 10** – This puzzle in the "Play Lab" lets the user create their own animation or game and share it with others. This puzzle can be repeated several times to create a new project every time. The sharing cards can be used to write down the URL to share with others.

Artist (Optional for second day)

The "Artist" lets the student create new designs using "move" and "turn" commands. The finished project can be shared like the Flappy Game is shared.

- **Puzzle 1** – A square is drawn using the "move" and "turn" blocks.
- **Puzzle 2** – A diamond is drawn by changing the angles in the "turn" blocks.
- **Puzzle 3** – A hexagon is drawn by turning 60 degrees for 6 times. The total trip theorem says the number of times turned times the angle turned will equal 360 in any polygon. Math at work in art.
- **Puzzle 4** – The "repeat number times" "do" loop is introduced to simplify the drawing of objects.
- **Puzzle 5** - A diamond flower is drawn using several "move" and "turn" blocks in a "repeat _____ times" block.
- **Puzzle 6** – The "set color value" is introduced to make the diamond flower prettier.
- **Puzzle 7** – The "repeat ___ times" block is used to create a hexagonal design.
- **Puzzle 8** - A function is another programming tool to help you avoid repeating yourself. This function draws a flower, so you can use it any time you want to draw a flower. Use this and the new "jump" block to draw flowers.
- **Puzzle 9** – The "draw a shape" function is introduced.
- **Puzzle 10** – Here a student can create their own function to draw a design and when finished it can be shared with others just like the "Flappy Game" is shared.

Additional Resources:

Videos

- Making the Invisible Visible in Video
- Robot Helps Kids with Autism Communicate
- Boston Dynamics Big Dog Robot

ScratchEd – Learn-Share-Connect

- ScratchEd is an online community for Scratch educators.

Scratch Creative Computing

- Scratch Creative Computing Guide
- Creative Computing PDF Manual

Scratch Game Tutorials

Using Scratch or Snap games can be created. The following YouTube videos show how this can be done.

- Meteor Dodge Game
- Reverse Gravity Game
- Catworm Game
- Shell Game
- Whack a Cat Game
- Block Breaker Game

Google CS First Clubs

Google's computer science education program, CS First, is designed to engage students in computer science. Teachers and community volunteers have facilitated clubs in school computer labs, community centers, and libraries. CS First materials are often used with students between 9 and 15 years old in an after-school setting, but some educators use the materials during the school day.

Google provides free CS First materials and loaner headphones to support clubs within the United States. All materials are also available online for International clubs. CS First clubs make students excited about and engaged with learning to code in a fun, supportive club atmosphere. The CS First curriculum was created by the Google Computer Science

Teaching Fellows, a cohort of graduates with degrees in computer science and education. The club materials provide everything students need to learn code, so no computer science experience is necessary to host a CS First club or to volunteer with students in a club.

Topics for CS First clubs include:

- Scratch Game Design
- Scratch Music and Sound
- Scratch Storytelling
- Scratch Fashion and Design
- Scratch Art
- Scratch Sports*
- Scratch Social*
- *coming soon

3rd Party Educator Resources

Tutorials for the Classroom:

- CodeHS—Online curriculum designed specifically for high school classrooms.
- Codecademy After School—A complete online afterschool program for a coding club.
- Khan Academy—Online curriculum that teaches JavaScript programming in a visual environment.
- Tynker—Teach programming in elementary or middle school in a fun way.

Curriculum You Can Integrate

- Bootstrap—Free curriculum to teach high-school algebra and geometry concepts using computer programming.
- CS Unplugged—Fun classroom exercises to teach computer science principles, no computers needed!
- Teach AP CS—Downloads, guides, videos, and tools to help teach AP Computer Science. Bring Classes to Your School.
- Globaloria—Get mentored and placed into high school classes as a part-time teacher in a team teaching model where the school district is unable to meet their students' computer science needs on its own.
- AP Computer Science - with Amplify—This online course (MOOC) is designed for high school students to learn AP Computer Science in an online classroom. The course is free, but additional support to the school offering it costs money.
- Teaching Kids Programming—Build the next generation of creators using free, open-source Java courseware. Platforms for Teaching Kids.
- Alice—3D programming environment that makes it easy to create an animation for telling a story, playing an interactive game, or a video to share on the Web.
- Kodu—Create games on the PC and XBox via a simple visual programming language. Can be used to teach creativity, problem solving, storytelling, as well as programming.
- Scratch—Programming language that makes it easy to create your own interactive stories, animations, games, music and art - and share your creations on the web.

Assessment(s):

- Explanation of concepts (written, oral, or through demonstration or performance of particular skills): The “Play Lab” lets a student write their own interactive story and animation.
- Critical thinking demonstration (written, oral, or through demonstration or performance): All of the projects involve computational thinking and problem-solving.
- At home project or activity related to the ILO or lesson objectives conducted with parents and reported by the student. All of the projects have URLs that can be shared with family, friends, and parents.
- Develop a portfolio of artifacts documenting concepts learned. All projects/artifacts can be linked to a portfolio.